TEMPORARY TABLE

A temporary table in SQL is a table that is created and used temporarily during a session. Temporary tables are useful for situations where you need to manipulate or organize data for complex queries or multiple-step operations but don’t need the table to persist permanently in the database.

Characteristics of Temporary Tables:

* Temporary tables exist only for the duration of the session or transaction.
* They are automatically dropped when the session ends or when explicitly dropped.

Syntax:

**Create temp table table\_name (column1, column2……...);**

VIEW

A view in PostgreSQL is a virtual table that represents the result of a query. It doesn't store data physically like a table, but instead stores a query, which gets executed whenever the view is called.

Syntax:

**CREATE VIEW view\_name AS**

**SELECT column1, column2, ...**

**FROM table\_name**

**WHERE condition;**

* All view are not updatable, a view containing limit and offset are not automatically updatable.
* View that do not select from a single table are not updatable.

Materialized View

A **materialized view** is a physical copy of the result set of a query stored in the database. It is precomputed and stored, making it faster for read-heavy operations, especially for complex queries or aggregations.

Syntax:

**CREATE MATERIALIZED VIEW materialized\_view\_name AS**

**SELECT column1, column2, ...**

**FROM table\_name**

**WHERE condition;**

* It does not automatically reflect changes in the underlying tables unless refreshed.

Refreshing a materialized view:

REFRESH MATERIALIZED VIEW materialized\_view\_name;

|  |  |  |
| --- | --- | --- |
| **Feature** | **View** | **Materialized View** |
| **Data Storage** | No physical storage (virtual) | Stores data physically |
| **Performance** | Slower for complex queries | Faster for repeated queries |
| **Data Freshness** | Always up-to-date | Requires manual or automatic refresh |
| **Complexity** | Simpler to create and maintain | More complex with refresh strategies. |

**Common Table Expression**

A Common Table Expression (CTE) in SQL is a temporary result set that is defined and used within the execution scope of a SELECT, INSERT, UPDATE or DELETE statement. CTEs are designed to make queries easier to read and maintain by allowing complex queries to be broken into modular components.

Syntax:

**With cte\_name as (**

**SELECT column1 , column2**

**FROM table\_name**

**WHERE condition**

**)**

**SELECT \* FROM cte\_name;**